Paul Johnson April 25, 2013

Data Management

```
library(foreign)
library(rockchalk)
i <- 43
dat <- read.dta(paste("../student-test2/student-",i,".dta", sep = ""))
```

The variables pprof and pnet are scored as numeric, but really they are factors. So convert them to prevent future mis-understandings.

```
dat$pprof <- factor(dat$pprof, labels = c("NO", "YES"))
dat$pnet <- factor(dat$pnet, labels = c("NO","YES"))
```

```
datsum <- summarize(dat)
```

Table would need some hand customization

```
library(xtable)
print(xtable(datsum$numeric, caption = "Best Automatic Summary Table for Numerics", label =
    "table1"), "latex")
```

	act	harv	ibs	sal1	sal2	sal3	sat
0%	7.77	1148.00	66.08	5204.00	7099.00	148200.00	1129.00
25%	19.01	1528.00	94.03	16540.00	19760.00	161200.00	1505.00
50%	22.24	1631.00	100.20	20150.00	23160.00	165500.00	1611.00
75%	25.87	1723.00	107.40	24160.00	27230.00	169800.00	1701.00
100%	38.54	2102.00	130.90	41180.00	41320.00	186400.00	2200.00
mean	22.29	1627.00	100.60	20400.00	23340.00	165500.00	1605.00
sd	5.14	155.00	10.25	5712.00	5863.00	5925.00	155.60
var	26.37	24010.00	105.10	32630000.00	34370000.00	35100000.00	24210.00
NA's	15.00	47.00	0.00	10.00	0.00	0.00	21.00
N	504.00	504.00	504.00	504.00	504.00	504.00	504.00

Table 1: Best Automatic Summary Table for Numerics

Let students figure way to beautify this:

```
print(datsum$factors)
```

```
              gender                      major                         pnet                         pprof
F             :259.0000    H             :170.000    NO            :349.0000    NO            :345
    .0000
M             :245.0000    S             :168.000    YES           :155.0000    YES           :159
    .0000
NA's          :  0.0000    N             :166.000    NA's          :  0.0000    NA's          :  0
    .0000
entropy       :  0.9994    NA's          :  0.000    entropy       :  0.8903    entropy       :  0
    .8994
normedEntropy:  0.9994     entropy       :  1.585    normedEntropy:  0.8903     normedEntropy:  0
    .8994
N             :504.0000    normedEntropy:  1.000     N             :504.0000    N             :504
    .0000
                           N             :504.000
```

Aptitude Test Variables

There's severe multicollinearity between the variables harv, sat, and act. It seems clear we can't estimate both sat and harv, and several students noticed that since harv is a summary of the other tests, then there's some reason to suppose sat is a better variable. (I know for a fact that harv = sat + act).

Please find Table 2. I left the Iowa Basic Skills variable in my best model, mainly because I wanted to estimate that coefficient, even though the F test below indicates one can exclude harv and ibs from the "full" model without losing any sleep.

```
m1s <- lm(sal1 ~ sat, data = dat)
m1a <- lm(sal1 ~ act, data = dat)
m1i <- lm(sal1 ~ ibs, data = dat)
m1h <- lm(sal1 ~ harv, data = dat)
m1all <- lm(sal1 ~ sat + act + ibs + harv, data = dat)
m1best <- lm(sal1 ~ sat + act + ibs, data = dat)
```

```
mcDiagnose(m1all)
```

```
The following auxiliary models are being estimated and returned in a list:
sat ~ act + ibs + harv
<environment: 0x1f815e8>
act ~ sat + ibs + harv
<environment: 0x1f815e8>
ibs ~ sat + act + harv
<environment: 0x1f815e8>
harv ~ sat + act + ibs
<environment: 0x1f815e8>
Drum roll please!

And your R_j Squareds are (auxiliary Rsq)
      sat       act       ibs      harv
0.9998411 0.8766747 0.2182352 0.9998443
The Corresponding VIF, 1/(1-R_j^2)
        sat         act         ibs        harv
6291.918669    8.108637    1.279157 6422.052275
Bivariate Correlations for design matrix
      sat  act  ibs harv
sat  1.00 0.30 0.38 1.00
act  0.30 1.00 0.37 0.33
ibs  0.38 0.37 1.00 0.39
harv 1.00 0.33 0.39 1.00
```

```
niceLabels <- c(act = "ACT", sat = "SAT", harv = "Harvard SS", ibs = "Iowa BS",  majorS = "
    Major: Soc.", majorN = "Major: Nat.", majorH = "Major: Hum.", pnetYES = "Parent Network
    : Yes",  pprofYES="Prof. Parents: Yes", genderM = "Gender: Male", "log(harv)"= "ln(
    Harvard SS)",
    "I(harv * harv)"= "Harvard SS$^2$", major2H = "Major 2: Hum.", major2N = "Major 2: Nat.
      ")
outreg(list(m1s, m1a, m1i, m1h, m1all, m1best), tight = TRUE, modelLabels = c("SAT","ACT","
    IBS","Harvard SS", "All", "Best"), varLabels = niceLabels, title = paste("Regression
    with sal1: Student-",i, sep=""), label = "tab:tab2")
```

Could conduct an F test of the hypothesis that $b_{ibs} = b_{harv} = 0$. But which model should I be testing? Test the one with all the variables, to see if *harv* and *ibs* should both be set to 0. To do that, I need to take the data frame used to fit m1all and use it to fit the restricted model. Otherwise, the F test fails.

```
m1alldf <- model.frame(m1all)
m1restricted <- lm(sal1 ~ sat + act, data = m1alldf)
anova(m1restricted, m1all)
```

```
Analysis of Variance Table

Model 1: sal1 ~ sat + act
Model 2: sal1 ~ sat + act + ibs + harv
```

Table 2: Regression with sal1: Student-43

	SAT Estimate (S.E.)	ACT Estimate (S.E.)	IBS Estimate (S.E.)	Harvard SS Estimate (S.E.)	All Estimate (S.E.)	Best Estimate (S.E.)
(Intercept)	-1463.054	12975.125*	10501.44*	-1239.285	-921.995	-2840.256
	(2540.846)	(1108.109)	(2492.023)	(2693.685)	(3191.594)	(2953.617)
SAT	13.631*	.	.	.	-127.908	11.724*
	(1.574)				(133.883)	(1.715)
ACT	.	333.622*	.	.	102.365	248.88*
		(48.415)			(144.373)	(51.963)
Iowa BS	.	.	98.372*	.	-24.867	-10.933
			(24.637)		(28.795)	(26.502)
Harvard SS	.	.	.	13.281*	139.384	.
				(1.646)	(133.771)	
N	473	479	494	448	416	458
RMSE	5319.267	5468.954	5627.51	5385.937	5303.81	5201.583
R^2	0.137	0.091	0.031	0.127	0.167	0.184
adj R^2	0.136	0.089	0.029	0.125	0.159	0.178

$*p \leq 0.05$

```
  Res.Df        RSS Df Sum of Sq      F Pr(>F)
1    413 1.1614e+10
2    411 1.1562e+10  2  52871828 0.9398 0.3916
```

Noticing this sample size problem, I wondered if I should re-do Table 2 so that all are fitted on the exact same data. Since I exclude harv, should those cases that are missing on harv "come back to life" when I exclude harv from the model? I think so. Still, there is something unappetizing about this. Fitting harv causes a loss of cases, no matter how we look at it. So for the best model and the ones for sat and ibs, I use the sample from the best model, but when harv enters the picture, we lose some cases.

```
m1best <- lm(sal1 ~ sat + act + ibs, data = dat)
dat2 <- model.frame(m1best)
m1s <- lm(sal1 ~ sat, data = dat2)
m1a <- lm(sal1 ~ act, data = dat2)
m1i <- lm(sal1 ~ ibs, data = dat2)
m1h <- lm(sal1 ~ harv, data = dat[row.names(dat2), ])
m1all <- lm(sal1 ~ sat + act + ibs + harv, data = dat[row.names(dat2), ])
```

```
outreg(list(m1s, m1a, m1i, m1h, m1all, m1best), tight = TRUE, modelLabels = c("SAT","ACT","
    IBS","Harvard SS", "All", "Best"), varLabels = niceLabels)
```

	SAT Estimate (S.E.)	ACT Estimate (S.E.)	IBS Estimate (S.E.)	Harvard SS Estimate (S.E.)	All Estimate (S.E.)	Best Estimate (S.E.)
(Intercept)	-1783.604	12791.03*	10235.84*	-1454.4	-921.995	-2840.256
	(2590.443)	(1139.632)	(2565.171)	(2780.891)	(3191.594)	(2953.617)
SAT	13.844*	.	.	.	-127.908	11.724*
	(1.605)				(133.883)	(1.715)
ACT	.	343.083*	.	.	102.365	248.88*
		(49.688)			(144.373)	(51.963)
Iowa BS	.	.	101.646*	.	-24.867	-10.933
			(25.368)		(28.795)	(26.502)
Harvard SS	.	.	.	13.435*	139.384	.
				(1.699)	(133.771)	
N	458	458	458	416	416	458
RMSE	5326.737	5466.319	5646.431	5397.829	5303.81	5201.583
R^2	0.14	0.095	0.034	0.131	0.167	0.184
adj R^2	0.138	0.093	0.032	0.129	0.159	0.178

$*p \leq 0.05$

Deciding what's "important"? We have lots of ways. If I've settled on a "best" model, it seems like I should be confined to the variables in that model. And the diagnostics should not depend on harv. Here are the partial and semi-partial correlations.

```
getPartialCor(m1best)
```

```
             sal1
sal1  -1.00000000
sat    0.30545009
act    0.21931389
ibs   -0.01935787
```

```
getDeltaRsquare(m1best)
```

```
The deltaR-square values: the change in the R-square
      observed when a single term is removed.
Same as the square of the 'semi-partial correlation coefficient'
    deltaRsquare
sat 0.0839851785
act 0.0412407135
ibs 0.0003059594
```

I admit, it is tough to conceptualize the scales of the different variables. I suppose I could scale the sat, act, and ibs scores so that they are all on the same 0-100 scale. Then I'll re-run the model. (This is called "percent of maximum" scoring (POMS)). Since we KNOW from previous work that re-scaling a variable has absolutely no substantive impact on the fit, and it is just for convenience of interpretation, this is an innocuous change.

```
dat2$satpoms <- 100*(dat2$sat - min(dat2$sat))/(max(dat2$sat) - min(dat2$sat))
dat2$actpoms <- 100*(dat2$act - min(dat2$act))/(max(dat2$act) - min(dat2$act))
dat2$ibspoms <- 100*(dat2$ibs - min(dat2$ibs))/(max(dat2$ibs) - min(dat2$ibs))
summarize(dat2[, c("satpoms","actpoms","ibspoms")])
```

```
$numerics
     actpoms ibspoms satpoms
0%      0.00    0.00    0.00
25%    36.70   42.94   35.21
50%    47.12   52.34   45.02
75%    59.17   63.99   53.32
100%  100.00  100.00  100.00
```

```
mean    47.39    53.23    44.57
sd      16.72    16.06    14.50
var    279.70   258.00   210.20
NA's     0.00     0.00     0.00
N      458.00   458.00   458.00

$factors
NULL
```

```
m1poms <- lm(sal1 ~ satpoms + actpoms + ibspoms, data = dat2)
summary(m1poms)
```

```
Call:
lm(formula = sal1 ~ satpoms + actpoms + ibspoms, data = dat2)

Residuals:
     Min       1Q   Median       3Q      Max
-17022.9  -3372.9     91.5   3317.6  18126.7

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 11611.706   1025.944  11.318  < 2e-16 ***
satpoms       125.542     18.368   6.835 2.65e-11 ***
actpoms        76.580     15.989   4.790 2.27e-06 ***
ibspoms        -7.087     17.179  -0.413     0.68
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 5202 on 454 degrees of freedom
Multiple R^2: 0.1838,  Adjusted R^2: 0.1784
F-statistic: 34.08 on 3 and 454 DF,  p-value: < 2.2e-16
```

Oh, one more thing. Recall my point that partial and semi-partial correlations are completely worthless when 1) there is multicollinearity and 2) we are uncertain which variables should be in consideration. Notice how crazy your conclusions would be if you based them on the "full" model.

```
options(scipen = 10)
getPartialCor(m1all)
```

```
            sal1
sal1 -1.00000000
sat  -0.04707298
act   0.03495271
ibs  -0.04255983
harv  0.05132858
```

```
getDeltaRsquare(m1all)
```

```
The deltaR-square values: the change in the R-square
      observed when a single term is removed.
Same as the square of the 'semi-partial correlation coefficient'
     deltaRsquare
sat   0.001849196
act   0.001018518
ibs   0.001510995
harv  0.002199582
```

```
options(scipen = 5)
```

Additional Variables

Please see Table 3 for the regressions.

Table 3: Regression with sal2: Student-43

	Test Scores Only Estimate (S.E.)	All Predictors Estimate (S.E.)
(Intercept)	1435.742	-1427.582
	(3048.339)	(2954.299)
SAT	11.879*	11.501*
	(1.771)	(1.697)
ACT	187.851*	239.257*
	(53.902)	(52.239)
Iowa BS	-12.872	-10.776
	(27.437)	(26.32)
Major: Soc.	.	1776.768*
		(588.514)
Major: Nat.	.	4033.181*
		(594.075)
Prof. Parents: Yes	.	642.333
		(520.207)
Parent Network: Yes	.	525.053
		(523.59)
Gender: Male	.	-365.298
		(482.106)
N	468	468
RMSE	5424.443	5189.018
R^2	0.148	0.229
adj R^2	0.143	0.216

$*p \leq 0.05$

```
m2small <- lm(sal2 ~ sat + act + ibs, data = dat)
m2all <- lm(sal2 ~ sat + act + ibs + major + pprof + pnet + gender, data = dat)
outreg(list(m2small, m2all), tight = TRUE, title = paste("Regression with sal2: Student-",i
    , sep=""),modelLabels = c("Test Scores Only","All Predictors"), varLabels = niceLabels,
     label = "table3")
```

Fancy T test. Lets use the big model to find out if $b_{pnetYES} = b_{pprofYES}$.

```
m2allc <- coef(m2all)
m2allv <- vcov(m2all)
numer <- m2allc["pprofYES"] - m2allc["pnetYES"]
names(numer) <- "difference"
denom <- sqrt(m2allv["pprofYES","pprofYES"] + m2allv["pnetYES","pnetYES"] - 2 * m2allv["
    pprofYES","pnetYES"])
print(paste("Fancy T: ", "Numerator = ", numer, "Denominator = ", denom))
```

```
[1] "Fancy T:  Numerator =  117.279797433104 Denominator =  708.670925512461"
```

```
tval <- numer/denom
print("T ratio is")
```

```
[1] "T ratio is"
```

```
tval
```

```
difference
 0.1654926
```

```
print("The two-tailed test would have p value")
```

```
[1] "The two-tailed test would have p value"
```

```
2 * pt(abs(tval), df = m2all$df, lower.tail = FALSE)
```

```
difference
 0.8686291
```

Could I make a function that "just" gets that right and would I be damaging students by ruining their educational experience? This would be very easy if the output had the variable names "pprof" and "pnet", but because I've made them factors, they are now pprofYES and pnetYES, and thus either my function has to be clever or the user's have to be clever in naming their request.

```
fancyT <- function(model, parm1, parm2){
     mc <- coef(model)
     mv <- vcov(model)
     numer <- mc[parm1] - mc[parm2]
     denom <- sqrt(mv[parm1, parm1]
         + mv[parm2, parm2] - 2 * mv[parm1, parm2])
     tval <- numer/denom
     tdf <- model$df
     tvalp <- 2 * pt(abs(tval), df = tdf, lower.tail = FALSE)
   res <- c(numer, denom, tval, tdf, tvalp)
   names(res) <- c("parm1 - parm2", "SE(parm1 - parm2)", "T", "df", "p-value")
   res
 }
fancyT(m2all, parm1 = "pprofYES", parm2 = "pnetYES")
```

```
    parm1 - parm2 SE(parm1 - parm2)                 T                df           p-value
      117.2797974       708.6709255         0.1654926       459.0000000         0.8686291
```

```
m2all <- lm(sal2 ~ sat + act + ibs + major + pprof + pnet + gender, data = dat)
m2alldf <- model.frame(m2all)
m2small <- lm(sal2 ~ sat + act + ibs, data = m2alldf)
anova(m2small, m2all)
```

```
Analysis of Variance Table

Model 1: sal2 ~ sat + act + ibs
Model 2: sal2 ~ sat + act + ibs + major + pprof + pnet + gender
  Res.Df         RSS Df  Sum of Sq      F         Pr(>F)
1    464 13653004178
2    459 12358989532  5 1294014646 9.6117 0.000000009729 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Nonlinear

```
nm1 <- lm(sal3 ~ harv + gender + major + pprof + pnet, data = dat)
nm2 <- lm(sal3 ~ log(harv) + gender + major + pprof + pnet, data = dat)
nm3 <- lm(sal3 ~ harv + I(harv*harv) + gender + major + pprof + pnet, data = dat)
library(rockchalk)
nd <- rockchalk::newdata(nm1, predVals = list(harv = plotSeq(dat$harv, 20)))
nd$m1fit <- predict(nm1, newdata = nd)
nd$m2fit <- predict(nm2, newdata = nd)
nd$m3fit <- predict(nm3, newdata = nd)
```

For the regression table, please see Table 4

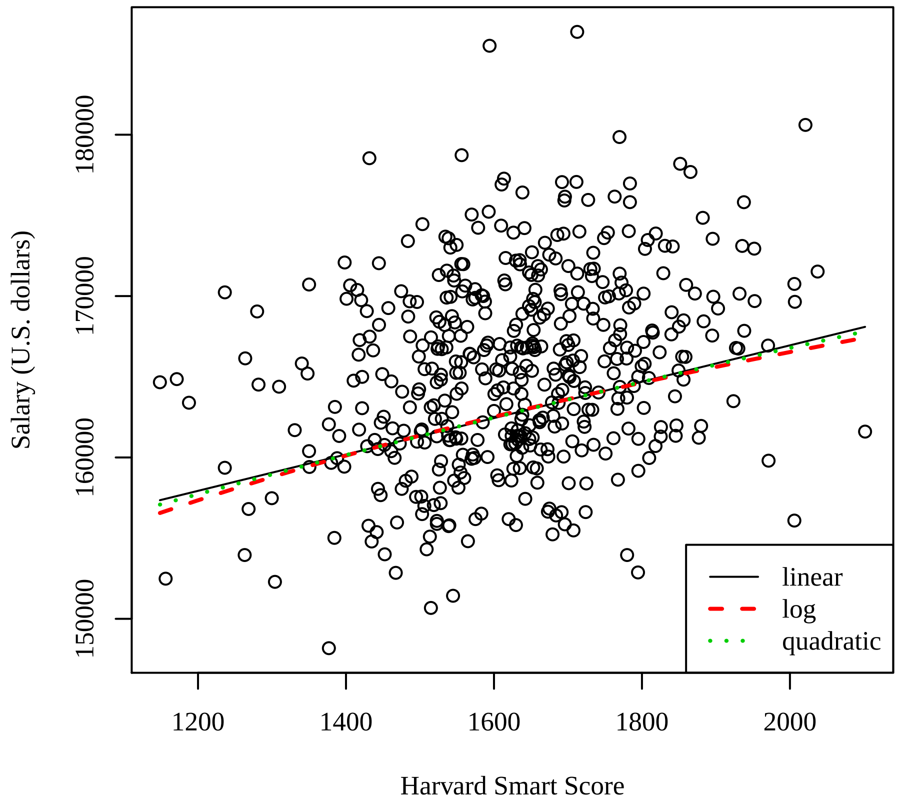
Table 4: Regression with sal3: Student-43

	Linear Estimate (S.E.)	Log Estimate (S.E.)	Quadratic Estimate (S.E.)
(Intercept)	144404.45*	29848.727	140911.937*
	(2573.706)	(18257.034)	(17261.706)
Harvard SS	11.272*	.	15.614
	(1.541)		(21.278)
Gender: Male	404.112	410.916	406.049
	(477.202)	(477.35)	(477.804)
Major: Soc.	1931.764*	1943.667*	1937.419*
	(582.101)	(582.374)	(583.377)
Major: Nat.	5658.953*	5656.096*	5659.53*
	(582.711)	(582.911)	(583.339)
Prof. Parents: Yes	1130.667*	1121.316*	1127.848*
	(519.143)	(519.272)	(519.879)
Parent Network: Yes	-1242.233*	-1232.117*	-1238.109*
	(520.952)	(521.13)	(521.896)
ln(Harvard SS)	.	17982.492*	.
		(2467.614)	
Harvard SS^2	.	.	-0.001
			(0.007)
N	457	457	457
RMSE	5087.734	5089.58	5093.159
R^2	0.259	0.258	0.259
adj R^2	0.249	0.248	0.247

$*p \leq 0.05$

```
outreg(list(nm1, nm2, nm3), tight = TRUE, title = paste("Regression with sal3: Student-",i,
    sep=""), modelLabels = c("Linear", "Log", "Quadratic"), varLabels = niceLabels, label
    = "table4")
```

```
plot(sal3 ~ harv, data = dat, xlab = "Harvard Smart Score", ylab = "Salary (U.S. dollars)")
lines(m1fit ~ harv, data = nd, lty = 1, col = 1)
lines(m2fit ~ harv, data = nd, lty = 2, col = 2, lwd = 2)
lines(m3fit ~ harv, data = nd, lty = 3, col = 3, lwd = 2)
legend("bottomright", legend = c("linear", "log", "quadratic"), lty = c(1,2,3), col = c
    (1,2,3), lwd = c(1,2,2))
```




```
cm1 <- lm(sal2 ~ major, data = dat)
dat$major2  <- relevel(dat$major, ref = "S")
cm2 <- lm(sal2 ~ major2, data = dat)
cm3 <- lm(sal2 ~ sat + act + ibs + major + pprof + pnet + gender, data = dat)
cm4 <- lm(sal2 ~ sat + act + ibs + major2 + pprof + pnet + gender, data = dat)
```

```
outreg(list(cm1, cm2, cm3, cm4), tight = TRUE, title = paste("Categorical Regressions:
    Student-", i, sep=""), modelLabels = c("major", "major2", "major full", "major2 full"),
     varLabels = niceLabels)
```

```
predictOMatic(cm1)
```

```
$major
              fit major
H (30%) 21651.91     H
S (30%) 23113.41     S
N (30%) 25306.50     N

attr(,"flnames")
[1] "major"
```

```
predictOMatic(cm2)
```

```
$major2
              fit major2
H (30%) 21651.91      H
S (30%) 23113.41      S
N (30%) 25306.50      N

attr(,"flnames")
[1] "major2"
```

Table 5: Categorical Regressions: Student-43

	major Estimate (S.E.)	major2 Estimate (S.E.)	major full Estimate (S.E.)	major2 full Estimate (S.E.)
(Intercept)	21651.912*	23113.409*	-1427.582	349.186
	(435.489)	(438.074)	(2954.299)	(2959.83)
Major: Soc.	1461.497*	.	1776.768*	.
	(617.705)		(588.514)	
Major: Nat.	3654.591*	.	4033.181*	.
	(619.574)		(594.075)	
Major 2: Hum.	.	-1461.497*	.	-1776.768*
		(617.705)		(588.514)
Major 2: Nat.	.	2193.093*	.	2256.413*
		(621.393)		(594.044)
SAT	.	.	11.501*	11.501*
			(1.697)	(1.697)
ACT	.	.	239.257*	239.257*
			(52.239)	(52.239)
Iowa BS	.	.	-10.776	-10.776
			(26.32)	(26.32)
Prof. Parents: Yes	.	.	642.333	642.333
			(520.207)	(520.207)
Parent Network: Yes	.	.	525.053	525.053
			(523.59)	(523.59)
Gender: Male	.	.	-365.298	-365.298
			(482.106)	(482.106)
N	504	504	468	468
RMSE	5678.087	5678.087	5189.018	5189.018
R^2	0.066	0.066	0.229	0.229
adj R^2	0.062	0.062	0.216	0.216

$*p \leq 0.05$